



## PATENT ABSTRACTS OF JAPAN

(11) Publication number: 05101054 A

(43) Date of publication of application: 23.04.93

(51) Int. Cl. G06F 15/20  
G06F 15/20

(21) Application number: 03285677

(71) Applicant: FUJITSU LTD

(22) Date of filing: 04.10.91

(72) Inventor: KANEDA KAZUKO

(54) DOCUMENT PREPARATION SUPPORTING  
SYSTEM BY SYNTHESIZING COMPONENT

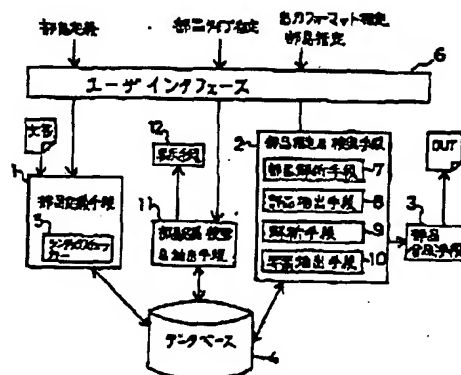
can be also supported to prepare the document and to  
divide the document into components.

## (57) Abstract

COPYRIGHT: (C)1993,JPO&amp;Japio

**PURPOSE:** To provide the document preparation supporting system for automatically preparing a document from the document, which is divided into component, in the case of preparing the document by dividing the document into the components and storing them in a database.

**CONSTITUTION:** The document is marked-up based on document definition decided in advance, and the document is divided into the components through a components defining means 1 and stored in a database 4. When the specifying of the document to be prepared is designated in the case of preparing the document, a component designating & extracting means 2 extracts required components from the database 4. A component synthesizing means 3 synthesizes the document from the retrieved components and prepares the document. The component defining means is provided with a syntax checker 5 and cracks whether mark-up is correct or not. On the other hand, a component definition retrieving & extracting means displays the document structure of the document stored in the database, and it





(19)日本国特許庁(J P)

(12) 公開特許公報(A)

(11)特許出願公開番号

特開平5-101054

(43)公開日 平成5年(1993)4月23日

(51)Int.Cl.<sup>5</sup>

G 0 6 F 15/20

識別記号

5 5 8

5 3 6

庁内整理番号

7343-5L

7343-5L

F I

技術表示箇所

審査請求 未請求 請求項の数5(全 23 頁)

(21)出願番号

特願平3-285677

(22)出願日

平成3年(1991)10月4日

(71)出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中1015番地

(72)発明者 金田 和子

神奈川県川崎市中原区上小田中1015番地

富士通株式会社内

(74)代理人 弁理士 京谷 四郎

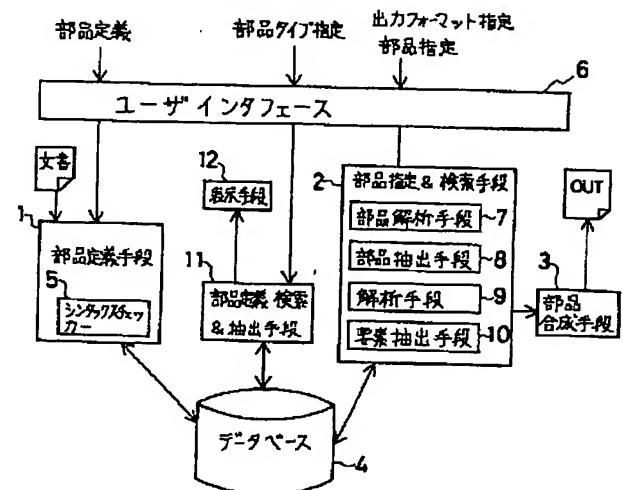
(54)【発明の名称】 部品合成による文書作成支援システム

(57)【要約】

本発明の原理ブロック図

【目的】 文書を部品化してデータベースに格納し、文書作成時に部品化された文書より文書を自動的に生成できる文書作成支援システムを提供すること。

【構成】 予め定めた文書定義に基づき文書にマークアップをして部品定義手段1を介して文書を部品化して、データベース4に格納する。文書作成時、作成する文書の仕様を指定すると、部品指定&抽出手段2は文書の作成に必要な部品をデータベース4より抽出する。部品合成手段3は検索された部品より、文書を合成し文書を作成する。部品定義手段は、シンタックスチェッカー5を備え、マークアップが正しいか否かチェックする。また、部品定義検索&抽出手段11により、データベースに格納されている文書の文書構造を表示し、文書の作成、文書の部品化をサポートすることもできる。



## 【特許請求の範囲】

【請求項1】 文書定義を定めてシステムに入力し、その定義を用いて構造化可能な文書にマークアップすることにより部品を定義する部品定義手段(1)と、定義された部品を格納するデータベース(4)と、

データベースに格納されている部品を指定し、指定された部品をデータベースより検索する部品指定&検索手段(2)と、

検索された部品を合成する部品合成手段(3)とを備え、文書作成時、データベース(4)に格納された部品を指定して、必要な部品を検索し、検索した部品を合成することにより文書を作成することを特徴とする部品合成による文書作成支援システム。

【請求項2】 文書定義を定めてシステムに入力し、その定義を用いて構造化可能な文書にマークアップすることにより部品を定義する部品定義手段(1)と、定義された部品を格納するデータベース(4)とを備えた文書作成支援システムにおいて、

マークアップされた文書の文書構造または構文とデータベースに格納されているマークアップ済文書の文書定義を対比するシンタックスチェッカー(5)を備え、シンタックスチェッカーによりマークアップされた文書の文書構造または構文をチェックすることを特徴とする部品合成による文書作成支援システム。

【請求項3】 文書定義を定めてシステムに入力し、その定義を用いて構造化可能な文書にマークアップすることにより部品を定義する部品定義手段(1)と、

定義された部品を格納するデータベース(4)と、部品ラベル名を入力する手段(6)と、

入力された部品ラベル名に基づき入力された部品を使用している文書名を解析する部品解析手段(7)と、

部品解析手段(7)により解析された文書名の文書をデータベースより検索し、入力された部品に相当する部分を抽出する部品抽出手段(8)と、

部品抽出手段(8)により抽出された部品を合成する部品合成手段(3)とを備え、

データベースに入力されている複数文書から所望の文書を作成することを特徴とする部品合成による文書作成支援システム。

【請求項4】 文書定義を定めてシステムに入力し、その定義を用いて構造化可能な文書にマークアップすることにより部品を定義する部品定義手段(1)と、

定義された部品を格納するデータベース(4)と、

文書の出力フォーマット定義を入力する手段(6)と、

入力された文書の出力フォーマット定義に基づき対応する部品を解析する解析手段(9)と、

解析手段(9)により解析された部品をデータベースより検索する要素抽出手段(10)と、要素抽出手段(10)により抽出された部品を合成する部品合成手段(3)とを備え、

データベースに入力されている文書から所望フォーマットの文書を作成することを特徴とする部品合成による文書作成支援システム。

トの文書を作成することを特徴とする部品合成による文書作成支援システム。

【請求項5】 文書定義を定めてシステムに入力し、その定義を用いて構造化可能な文書にマークアップすることにより部品を定義する部品定義手段(1)と、定義された部品を格納するデータベース(4)とを備えた文書作成支援システムにおいて、

入力された部品タイプに応じて、データベースに格納されているマークアップ済文書の部品定義を抽出する部品定義検索&抽出手段(11)と、抽出された部品定義を表示する表示手段(12)とを備え、

所望の文書定義に基づいて新しい文書または部品を作成する際、入力された部品のタイプに基づき部品定義検索&抽出手段(11)により部品定義を抽出し、抽出された部品定義を表示手段(12)により表示することにより、部品の作成を支援することを特徴とする部品合成による文書作成支援システム。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】 本発明は部品合成による文書作成支援システムに関する。一般に文書の作成においては、前に作成した文書の一部を再利用して新たに文書を作成したり、複数文書から1つの文書を作成したり、また、1つの情報を別の文書スタイルで作成していくといった場合が多い。プログラミングの場合は、再利用や合成を行う際、プログラムをブロック化し、ブロック毎にプログラムを部品として保持しておき、プログラミングに際して、必要なブロックを検索し合成をするといった方法がとられている。

【0002】 文書の作成においても、例えばビジネス文書、手引き書、手紙などのように構造化することが可能な文書に対して、文書を構造化し、そのブロックを再利用可能な部品として扱う機能、また、その部品を自動的に検索し合成する機能を持つシステムが望まれる。本発明は、上記のように構造化可能な文書に対して、文書を構成する部品を定義し、部品を指定し、部品を検索し、部品を合成する機能をもつ文書作成を支援するシステムに関するものである。

## 【0003】

【従来の技術】 従来、プログラムの作成において、プログラムをブロックに切りわけ、それらの部品を合成することにより新たなプログラムを作成することなどが考えられてきた。この場合におけるプログラムの部品定義、即ち、部品化は、ソースコードを部品化し、その骨組みに相当する部品と、その骨組みに埋め込まれる部分部品を組として切り出す方法や、また、ある処理の結果が次に実行する処理の入力データとなるようにプログラムを切りわけ、それらの処理を部品として切り出す方法などが考えられる。

【0004】 また、部品を指定する場合は、ユーザーが

どのような部品をどのような順番で組み合わせるかを指定したり、また、骨組みの部品を指定したりといった方法がとられる。さらに、部品の検索、合成においては、ユーザーが定める仕様に依りてユーザーが部品を選び、骨組みにあわせて部品を埋め込んでいくといった処理を行うことが考えられる。

【0005】一般の文書においては、文書の文脈自体を部品として保持し、これを用いて文書を作成することも考えられるが、一つの事項を文書化する場合でもその言い表し方は非常に多く、それらを1文書内で統一することは困難である。したがって、1文書内における言い表し方まで考慮する場合には、文書に何の修正を行わずにそのまま部品化することはできず、各部品ごとにその必要に応じて修正する作業が必要になってくる。

【0006】また、文書の作成においては、文書間のつながりを意識しなければならないが、これらのつながりを部品化することは困難である。以上のように、一般の文書を部品化し、これらの部品を検索、合成して文書を作成することは、種々の問題点があり困難であった。

【0007】そこで、例えばビジネス文書、手引き書、手紙などのような構造化可能な文書を対象として、文書の部品化を行うことが考えられる。しかし、前述したように文脈を部品化すると、プログラムの部品化のように骨組みに相当する部品とその骨組みに埋め込まれる部分部品を組として部品化し保持するのでは、部品を利用する際の部品の指定の仕方が面倒であるし、また、この骨組みにはこの部品といったようにその組み合わせを決められてしまうと、その再利用がしにくくなる。また、複数人で作成した文書から1つの文書を作成する場合にも、文書の作成者が、予め作成され部品化された文書のうち、どの文書のどの部分を用いるかを指定し選択することは容易ではないし、また、システム側で検索することも難しいという問題がある。以上のように文書を部品化し、その部品を合成し文書を作成することは種々の問題があり、いまだ実用化されているものがなかった。

【0008】

【発明が解決しようとする課題】本発明は、上記従来技術の問題点に鑑み、文書の内、部品化された文書に何の修正を加えることなくそのまま利用できる、例えばビジネス文書、手引き書、手紙などのように構造化可能な文書を対象とし、これらの文書を容易に部品化でき、また、新たな文書作成時に部品化された文書を容易に指定できるようにするとともに、部品の一部検索が可能であつて、部品を合成して新たな文書を作成する際、文書を自動的に生成できる文書作成支援システムを提供することを目的とする。

【0009】

【課題を解決するための手段】本発明は上記課題を解決するため、図1に示すように、構造化可能な文書を修正を加えることなくそのままの形で構造化して部品として

データベースに格納し、その部品を用いて文書を作成する文書作成支援システムを構成したものであつて、請求項1は、文書定義を定めてシステムに入力し、その定義を用いて構造化可能な文書にマークアップすることにより部品を定義する部品定義手段1と、定義された部品を格納するデータベース4と、データベースに格納されている部品を指定し、指定された部品をデータベースより検索する部品指定&検索手段2と、検索された部品を合成する部品合成手段3とを備えている。そして、文書作成時、データベース4に格納された部品を指定して、必要な部品を検索し、検索した部品を合成することにより文書を作成する。請求項2は、文書定義を定めてシステムに入力し、その定義を用いて構造化可能な文書にマークアップすることにより部品を定義する部品定義手段1と、定義された部品を格納するデータベース4とを備えた文書作成支援システムにおいて、マークアップされた文書の文書構造または構文とデータベース4に格納されているマークアップ済文書の文書定義を対比するシンタックスチェッカー5を備えている。そして、シンタックスチェッカー5によりマークアップされた文書の文書構造または構文をチェックする。請求項3は、文書定義を定めてシステムに入力し、その定義を用いて構造化可能な文書にマークアップすることにより部品を定義する部品定義手段1と、定義された部品を格納するデータベース4と、部品ラベル名を入力する手段6と、入力された部品ラベル名に基づき入力された部品を使用している文書名を解析する部品解析手段7と、部品解析手段7により解析された文書名の文書をデータベース4より検索し、入力された部品に相当する部分を抽出する部品抽出手段8と、部品抽出手段8により抽出された部品を合成する部品合成手段3とを備えている。そして、データベース4に入力されている複数文書を統合して指定した部品を用いた所望の文書を作成する。請求項4は、文書定義を定めてシステムに入力し、その定義を用いて構造化可能な文書にマークアップすることにより部品を定義する部品定義手段1と、定義された部品を格納するデータベース4と、文書の出力フォーマット定義を入力する手段6と、入力された文書の出力フォーマット定義に基づき対応する部品を解析する解析手段9と、解析手段により解析された部品をデータベース4より検索する要素抽出手段10と、要素抽出手段10により抽出された部品を合成する部品合成手段3とを備えている。そして、データベース4に入力されている文書から所望フォーマットの文書を作成する。請求項5は、文書定義を定めてシステムに入力し、その定義を用いて構造化可能な文書にマークアップすることにより部品を定義する部品定義手段1と、定義された部品を格納するデータベース4とを備えた文書作成支援システムにおいて、入力された部品タイプに応じて、データベース4に格納されているマークアップ済文書の部品定義を抽出する部品定義検索&抽出

手段11と、抽出された部品定義を表示する表示手段12とを備えている。そして、所望の文書定義に基づいて新しい文書または部品を作成する際、入力された部品のタイプに基づき部品定義検索&抽出手段11により部品定義を抽出し、抽出された部品定義を表示手段12により表示することにより、部品の作成を支援する。

#### 【0010】

【作用】請求項1のものにおいて、ユーザーは、部品定義手段1により、文書定義を定めてシステムに入力し、その定義を用いて構造化可能な文書にマークアップすることにより部品を定義しデータベース4に格納しておく。文書作成時、ユーザーはユーザインタフェース6より、作成する文書の仕様を入力する。部品指定&検索手段2はユーザーの仕様に応じた部品をデータベース4より検索する。検索された部品は部品合成手段3において合成され、出力される。請求項2のものにおいて、文書を部品化する場合、ユーザーは、文書定義に基づき構造化可能な文書にマークアップし、ユーザインタフェース6を介して部品定義手段1に与える。部品定義手段1は、マークアップされた文書の文書構造または構文とデータベース4に格納されているマークアップ済文書の文書定義をシンタックスチェッカー5により対比し、マークアップされた文書の文書構造または構文をチェックする。請求項3のものにおいて、ユーザーがデータベース4に入力されている複数文書を統合して指定した部品を用いた所望の文書を作成する場合、ユーザインタフェース6を介して、部品を指定する。部品解析手段7は入力された部品ラベル名に基づき入力された部品を使用している文書名を解析し、部品抽出手段8はデータベースより解析された文書名の文書を検索し、入力された部品に相当する部分を抽出する。部品合成手段3は抽出された部品に相当する部分を合成し、統合された文書を出力する。請求項4のものにおいて、ユーザーがデータベース4に入力されている文書から所望フォーマットの文書を作成する場合、ユーザインタフェース6を介して文書の出力フォーマット定義を入力する。解析手段9は入力された文書の出力フォーマット定義に基づき対応する部品を解析し、要素抽出手段10は解析された部品をデータベースより検索する。部品合成手段3は要素抽出手段10により抽出された部品を合成し、所望のフォーマットの文書を出力する。請求項5のものにおいて、ユーザーが、所望の文書定義に基づいて新しい文書または部品を作成する場合などにおいて部品定義を参照する際、ユーザインタフェース6を介して参照したい部品タイプを入力する。部品定義検索&抽出手段11は入力された部品タイプに応じて、データベース4に格納されているマークアップ済文書の部品定義を抽出し、表示手段12により表示させる。

#### 【0011】

【実施例】図2は本発明の文書作成支援システムの全体

構成を示す図である。同図において21はユーザインタフェース、22は部品定義手段、23はデータベース、24は部品指定&検索手段、25は部品合成手段、26はレイアウト、27は出力文書である。

【0012】図2におけるユーザインタフェース21はユーザーとシステムとの間のインタフェースをとる手段であり、ユーザーはユーザインタフェース21を介して、文書の部品化するためのマークアップあるいは、作成文書仕様の入力を行う。部品定義手段22は文書定義に基づきマークアップされた構造化可能な文書を部品定義（部品化）する手段、データベース23は部品化された文書を格納する手段、部品指定&検索手段24はユーザーが定める仕様に基づき部品を指定し、指定した部品をデータベースから検索する手段、部品合成手段25はデータベースから検索された部品を合成し文書を作成する手段、レイアウト26は部品合成手段25によって合成された文書をレイアウトする手段、出力文書27は合成されレイアウトされた文書である。

【0013】次に図2のシステムによる文書の作成手順を説明する。まず、ユーザーは構造化可能な文書の文書タイプに応じて文書定義を記述し、文書にマークアップを行う。マークアップとは定められた定義に基づき文書に、例えば図3に示すような<手引書>、<サブシステム>などのタグを付すことであって、タグが付された文書はシステムの文書定義手段22において上記定義に基づき階層化され、部品としてデータベース23に格納される。

【0014】なお、マークアップ機能は既存の機能であり、例えば電子文書交換をスムーズに行うために使用されている記述言語としてSGML（Standard Generalized Markup Language）（この言語による文書構造記述はISO国際標準化機構のSC18/WG8が担当）が知られている。この記述言語は文書構造を定義し、これに文書内容をマッピングするためのものであつて、文書としては新聞、辞書、メモ、手紙などあらゆるものがその対象となる。上記記述言語において、<\*>や</\*>はタグを表し、タグは文書要素、例えば章や見出し等の始まりと終わりを区切り、また、その種別を表す識別子でもある。

【0015】文書を部品化するというのは、文書の論理構造を階層的に定め、その各ブロックを部品とすることであつて、例えば、図3のようにマークアップされた手引書は図4に示すように階層化され、「サブシステム」、「コマンド」などの各ノードで区切られたブロック、例えば、「名」、「概要」などが部品となる。また、「手引書」などの文書タイプの名前はキーとしてデータベース23に格納されており、構造化可能な文書タイプがどのような部品で構成されているかという定義も部品と同様データベース23に格納されている。

【0016】図5は部品のデータベースへの格納形態を

示す図である。データベース内の部品は既存の文書定義であるSGMLのDTDで記述されている。DTD(Document Type Definition)とは前述したSGMLの文書構造をSGMLのシンタックスで記述したものである。すなわち、`<!DOCTYPE....>`で文書のタイプを定義しており、図5において、`<!DOCTYPE手引書*`は文書のタイプとして「手引書」を定義している。

【0017】また、文書に対する論理構造は`<!ELEMENT....>`の形式で定義していく。図5の例においては、「手引書」は「サブシステム」の集合からなり、「サブシステム」は「名」、「概要」、「コマンド」などから構成され、「コマンド」は「形式」、「説明」、「引数」、「注意」、「例」などから構成されていることを示している。これらの構造は、図4に示すように構文木で表される。

【0018】さらに、例えば、「注意」、「例」などに付された\*は0回以上の繰り返しを意味する。すなわち、「注意」、「例」などは無い場合もあるし、また、複数記載されている場合もあり、これらのものには\*が付される。またさらに、例えば「サブシステム」に付された+は1回以上の繰り返しを意味する。すなわち、手引書の中には少なくとも1以上のサブシステムが存在しており、これらのノードには+が付される。

【0019】以上のように文書をマークアップして部品化しデータベース23に格納したのち、ユーザーはユーザインタフェース21を介して、作成しようとする文書の仕様を入力し、所望の文書を作成する。例えば、ユーザーがデータベース23に格納された複数の文書をまとめて一文書を作成したり、また、ユーザーがデータベース23に格納された文書をその文書とは異なったフォーマットの文書に変換する場合、これらの仕様をユーザインタフェース21を介して入力する。

【0020】部品指定&検索手段24はユーザーの与えた仕様に応じてデータベース23より必要な部品を検索し、部品合成手段25に与える。部品合成手段25はデータベース23より検索された部品を合成して文書を作成する。部品合成手段25により作成された文書は必要に応じてレイアウト26によりレイアウトされ、文書27として出力される。

【0021】レイアウトする手段としてはTeX(AMS(American Mathematical Society)の商標)が知られている。TeXには文書のレイアウトや、数式の記述、表を作成するコマンドが豊富にあり、ユーザーの目的に合わせて文書整形を可能としている。TEXには色々の種類があり、LATEX、JTEX(NTT版)、JLATEX(ASCII版)などがある。

【0022】図6は本発明の図2における部品定義手段22の1実施例を示す図であり、本実施例は、ユーザーのマークアップが正しく行われたか否かをチェックする

機能を図2の文書定義手段22に設けたものである。同図において、31はユーザインタフェース、32は文書定義の対象となる文書、33はデータベース、34はマークアップ機能、35はマークアップされた文書を格納するファイルであって、例えば前記した図3に示されるマークアップ済の文書が格納される。また、データベース33内には、構造化可能な複数の文書が予め部品化されて、例えば図5に示したような形態で格納されている。さらに、36はシンタックスチェッカー、37は解析手段、38は差分検出手段である。

【0023】図6におけるユーザインタフェース31は図2において説明したようにシステムとユーザー間のインタフェースをとる手段であって、本実施例においては、ユーザーはユーザインタフェース31を介して文書のマークアップを行う。マークアップ機能34は前述したように、ユーザーのマークアップ操作に応じてファイルなどに格納された文書32にマークアップを行う手段、ファイル35はマークアップされた文書を格納するファイル、データベース33は部品定義された文書が格納されているファイルである。また、シンタックスチェッカー36はマークアップ済文書のマークアップをチェックする手段であり、その解析手段37はマークアップされた文書およびデータベースに格納された文書より構文木を作成し、差分検出手段38はマークアップ済文書の構文木とデータベースに格納された文書の構文木の差分を検出する手段である。

【0024】図7は本実施例における処理を説明する図であり、図7を用いて、図6のシステムの処理を説明する。まず、ユーザーは前述したように、既存のマークアップ機能73(図6の34)を用いて、文書72(図6の32)をマークアップする。すなわち、文書72に、前述の図3に示したように、`<手引き書>`、`<サブシステム>`などのタグを付けて、マークアップ済文書74を作成する。マークアップ済文書は図6の文書ファイル35に格納する。

【0025】シンタックスチェッカー78の解析手段37(図6)はマークアップ済文書74から構文木を作成するとともに、マークアップ済文書74の文書タイプの名前をキーとしてデータベース77(図6の22)よりその文書タイプを定義している構文を取り出し構文木76に変換する。すなわち、この文書が手引書を対象としていることから、データベース77より「手引書」を定義している構文を検索し、その部分を構文木76に変換する。

【0026】ついで、シンタックスチェッカー78の差分検出手段38(図6)は、構文木75と構文木76の対応するノードにおいてその子ノードを比較してその差分を検出し、マークアップ済文書74のエラーを検出する。この例においては、構文木76に対して構文木75には「引数」および「注意」が無いという結果が得られ

る。しかし、前述したように、「注意」には\*が付されており、これは「注意」のノードは無くても良いことを意味しているから構文木75に「注意」がないことはエラーにはならない。結局、「コマンド」定義のところでは「引数」の定義が抜けていることから、「<引数>マークがない」がシンタックスチェッカー78の出力79にエラーとして表示される。

【0027】図8は本実施例のシンタックスチェッカーのフローチャートであり、ステップS1においてマークアップを行ったのち、マークアップされた文書から構文木を抽出する(ステップS2)。一方、マークアップされた文書から文書タイプを抽出し(ステップS3)、次いで、文書タイプを定義している部分をデータベースから抽出し(ステップS4)、定義の構造を構文木として抽出する(ステップS5)。ステップS6において、ステップS2およびステップS5で得られた構文木を比較し、その差分が表示すべき差分であるか否かを文書定義により解析し(ステップS7)、差分を表示する(ステップS8)。

【0028】以上に説明したように、図6の実施例においては、ユーザーがマークアップした文書を予めデータベースに格納されている文書定義と比較することにより、文書の構文や構造の間違いを容易に検出できるので、文書の部品定義における処理効率を向上することができる。

【0029】図9は本発明の文書作成における第1の機能の実施例であり、本実施例は、部品を指定してその部品を構成要素として持つ文書を検索することにより、複数文書を統合し、1文書を作成する機能の実施例である。例えば、ソフトの開発段階で、「コマンド」を説明する複数の文書が作成された場合、これら「コマンド」の説明文書を部品化してデータベースに格納しておき、必要に応じて、これらの文書を統合して「コマンド」一覧を作成する場合などに用いることができる。

【0030】同図において41はユーザインタフェース、42は部品指定手段、43は解析手段、44は部品抽出手段、45は部品合成手段、46は部品情報のテーブル、47はデータベースである。ユーザインタフェース41は図2において説明したようにシステムとユーザー間のインタフェースをとる手段であって、本実施例においては、ユーザーはユーザインタフェース41を介して、ユーザーが例えば「コマンド」の一覧文書の作成をシステムに要求し、システム側からは作成された文書が出力される。部品指定手段42はユーザーの文書作成要求に応じて必要な部品を指定する手段、解析手段43は部品情報のテーブル46を参照して、指定された部品がどの文書で用いられているかを解析する手段、部品抽出手段44は解析手段43において求めた文書より必要な部品をデータベース47より抽出する手段、部品合成手段45は抽出された部品を合成して文書を作成する手段

である。

【0031】図10は本実施例における処理を説明する図であり、図10を用いて、図9のシステムの処理を説明する。ユーザーがデータベース47(図9)に格納されている複数文書の部品、例えば「コマンド」の部品だけを集めて文書を作成しようとする場合、ユーザーはユーザインタフェース81(図9の41)を介して、「コマンド」一覧の作成要求を行う。

【0032】解析手段83は「コマンド」部品がどの文書で使用されているかを部品情報テーブル84をもとに検索する。部品情報テーブル84は、例えば「buhin1」、「buhin2」などの文書名と、例えば「コマンド」などの部品ラベル名をキーとしてそれぞれの情報を検索できるように構成されており、この例においては「コマンド」をキーとして、それがどの文書で使用されているかを検出することができる。

【0033】この例においては、「buhin1」という文書1および「buhin2」という文書2で「コマンド」を使用しているので、文書1および文書2がデータベース47(図9)より検索される。抽出手段85

(図9における44)は文書1および文書2から図10の88に示す構文木を作成し、その構文木の「コマンド」のノードの部分だけを抽出する。すなわち、図10のデータベースにおける<BUHIN1>86および<BUHIN2>87の<コマンド>と</コマンド>で挟まれた範囲の内容を抽出する。

【0034】抽出された「コマンド」の部分は部品合成手段89(図9の45)で合成され、1つの文書に合成された「コマンド」一覧がユーザインタフェース81

(図9の41)より出力される。図11は本実施例のフローチャートであり、ユーザーが部品Eの指定を行うと(ステップS11)、テーブルにより、指定された部品Eを構成要素とするすべての部品(例えばP)を検索する(ステップS12)。次いで、検索された部品Pをデータベースより抽出し(ステップS13)、その部品Pから指定された部品Eの部分抽出する(ステップS14)。以上の手順により全ての部品(文書)から部品Eを構成要素とする部分を抽出したのち、抽出された部分を合成する(ステップS15)。

【0035】図12は本発明の文書作成における第2の機能の実施例であり、本実施例は、ユーザーがデータベース23に格納された文書をその文書とは異なったフォーマットの文書に変換する機能の実施例である。例えば、ソフトの開発段階で使われた設計資料等を部品化してデータベースに格納しておき、その必要部分から所望の書式の文書(以下の実施例においては「ライブラリ」の例で説明されている)を作成する場合などに用いることができる。本実施例の機能により、一つの情報を種々の書式の文書の中で用いることができるので、文書作成の効率を向上することができる。



【0036】図12において、51はユーザインタフェース、52はユーザーが与える出力フォーマットの定義、53は解析手段、54は要素抽出手段、55はデータベース、56は部品合成手段、57はレイアウト、58は出力文書である。

【0037】図12におけるユーザインタフェース51は前述したようにシステムとユーザー間のインタフェースをとる手段であって、本実施例においては、ユーザーはユーザインタフェース51を介して、文書の出力フォーマット定義52をシステムに与えて文書の作成をシステムに要求し、システム側からは作成された文書が出力される。解析手段53はユーザーが与えた出力フォーマット定義を解析し、所望の文書を作成するために必要な部品を定める手段、要素抽出手段54は解析手段53により定められた部品よりデータベース55内の要素を抽出する手段、部品合成手段56は要素抽出手段54によって抽出された要素を合成し、文書を作成する手段、レイアウト57は合成された文書をレイアウトし出力文書58を出力する手段である。データベース55にはソフト開発段階などで作成された文書が部品として格納されており、データベース内の部品には1つ上の部品を属性情報（例えばクラス1はサブシステム1に属する等の情報）として持たしておく。

【0038】図13は本実施例における処理を説明する図である。ここで、図13における出力フォーマット定義92の例は図14に示されており、また、部品合成されたマークアップ済文書99の例は図15に示されており、さらに、出力文書の例は図16に示されている。またさらに、図17は図14を構文木に変換したものである。次に、図13ないし図17を用いて、図12のシステムの処理を説明する。部品化された設計資料などの情報に基づき所望の書式の文書を作成する場合、ユーザーはユーザインタフェース91（図12の51）を介して作成したい文書の出力フォーマットの定義を入力する。例えば、「ライブラリ」という書式の文書を作成する場合、ユーザーは「ライブラリ」の出力フォーマットの定義（図14参照）を入力する。

【0039】ライブラリという定義が入力されると、解析手段93（図12の53）はその定義に記述されるプリミティブなものを抽出する。プリミティブなものとは、構文定義を展開していった時、展開できなくなったものを意味し、例えば図14の定義の場合は、「概要」、「形式」、「説明」などである。また、上記定義を構文木に変換し、そのリーフを抽出することも可能である。例えば、上記構文定義を構文木で表現すると図17で表されるが、この構文木における「概要」、「形式」、「説明」がリーフに相当する。

【0040】要素抽出部94（図10における54）は上記のようにして抽出された部品とその1つ上の部品をキーとしてデータベース95（図12における55）内

の要素を抽出する。例えば、図14の定義の場合には、まず、2行目の「概要」がプリミティブなものとして取り出され、この「概要」と図14における1つ上の部品である「ライブラリ」をキーとして、「概要」の内容がデータベース95から取り出される。その結果、図15に示すa、すなわち「ライブラリアンの概要が書かれます。」が取り出される。

【0041】次に、図14の3行目の「概要」がプリミティブなものとして取り出される。ここで、上記「概要」の1つ上の部品である「サブシステム」には「+」が付されており、これは「サブシステム」が1回以上繰り返されることを意味しているから、「サブシステム」とその上の部品である「ライブラリ」をキーとして抽出されるものが全てとりだされる。また、「サブシステム」がプリミティブなもので無い場合には、抽出されたサブシステムの「名」が親のキーとなる。

【0042】この例においては、＜ライブラリ＞、＜サブシステム＞で図15に示すb、すなわちサブシステムの名前「ライブラリアン」が取り出され、この＜ライブラリアン＞が親のキーとなって、＜ライブラリアン＞と＜概要＞で図15に示すc、すなわち「ライブラリアンの概要が書かれます。」が取り出される。上記のように、上の階層から順次、下層レベルまで同様の処理を行い、必要な部品を抽出していく。

【0043】部品合成手段96（図12の56）では、抽出されたものを合成し、マークアップを施し、前述した図13に示すSMGL文書が作成する。レイアウト97（図12における57）はSGML文書と前述したLaTeXのコマンドを対応させる定義を作成し、自動的にLaTeXコマンドを組み込ませることにより、図15に示すSMGL文書をLaTeXの組版プログラムに変換する。この組版プログラムをLaTeXプロセッサにかけると、フォーマットされた図16に示す文書が出力される。

【0044】図18は本実施例のフローチャートであり、ユーザーが出力フォーマット定義を入力すると（ステップS21）、そのフォーマット定義を解析し（ステップS22）、その解析結果に基づき、予め例えばソフトのドキュメント等のある目的に従った文書の各部分が部品として格納されているデータベースより、必要な部品を検索する（ステップS23）。次いで、検索された部品を抽出し（ステップS24）、フォーマット定義に従い、抽出された部品を合成し、出力する（ステップS25、ステップS26）。

【0045】図19は本発明の文書作成支援システムに設けることができる、文書のデータ構造を表示する機能の1実施例である。例えば、ユーザーが新しい文書を作成したり、また、文書を部品化する場合、データベースに格納されている文書のデータ構造を参照したい場合がある。本実施例はこの様な要求に答えるための機能であ

って、本実施例の機能を用いることにより、所望の文書タイプを検索し、そのタグのみを表示させることができる。図16において、62はユーザインタフェース、63は部品定義検索&抽出手段、64はデータベース、65は表示手段である。

【0046】図19におけるユーザインタフェース62は前述したようにシステムとユーザー間のインタフェースをとる手段であって、本実施例においては、ユーザーはユーザインタフェース62を介して、文書のタイプを指定して所望のタイプのタグの表示をシステムに要求し、システム側はその要求に応じて、ユーザインタフェース62を介して文書のタグを表示する。部品定義検索&抽出手段63は要求された文書のタイプに応じてデータベース64よりそのタイプを定義している部分を抽出し、それを構文木に変換する手段であり、表示手段65は部品定義検索&抽出手段63により抽出された構文木のノードをタグに変換し、変換されたタグを表示する手段である。なお、データベース64には前記した図5の格納形態で部品が格納されている。

【0047】図20は本実施例における処理を説明する図であり、図20を用いて、図19のシステムの処理を説明する。ユーザーはある文書定義に従って新しい文書を作成したり、部品を作成したりする場合、まず、ユーザーは文書のタイプを指定する。例えば、「手引書」の文書定義に基づいて新しい「手引書」の文書を作成する場合、ユーザー101はユーザインターフェイス102（図19の62）を介して「手引書」作成と指定する。「手引書」の指定がされると、部品定義検索&抽出手段103（図19の63）は「手引書」をキーとして、データベース104（図19の64）より「手引書」のタイプ定義がなされている部分を検索する。すなわち、データベースにおいて文書定義として<!DOCTYPE手引書...の構文を検索し、構文木106に変換する。

【0048】ついで、変換された構文木106から各ノードを抽出する。各ノードは、スタートタグとエンドタグが必要であるとか、エンドタグは無くてもよいとかの情報、親子関係、順序関係の情報を持っており、抽出する際、例えば、「手引書」には「サブシステム」の子ノードがあり、「サブシステム」には、「名」、「概要」、「コマンド」の子ノードがあること、また、「手引書」のタグは</手引書>のエンドタグが必要である等の親子関係、順序関係の情報も一緒に抽出する。

【0049】表示手段105は、データベースから検索され抽出されたノードをタグの形に変換し、親子関係、順序関係の情報を基に、107に示すように、<手引書>のスタートタグの次に<サブシステム>のスタートタグ、最後に「手引き書」のエンドタグとして</手引書>がくるといった関係を<手引書>のテンプレートとして表示する。

【0050】図21は本実施例のフローチャートであ

り、ユーザーが文書タイプの指定を行うと（ステップS31）、データベースより文書タイプを定義している部分を抽出し（ステップS32）、それより、構文木を作る（ステップS33）。次いで、構文木よりノードやリーフを抽出し（ステップS34）、定義に従いこれらをタグに変換し（ステップS35）、その情報を基にテンプレートを表示する（ステップS36）。

【0051】本実施例の機能を用いることにより、所望のタイプの文書のタグがテンプレートとして表示されるので、文書のマークアップ等の容易化など、ユーザーによる新しい文書の作成、文書の部品化を支援することができる。

#### 【0052】

【発明の効果】以上説明から明らかなように、本発明によれば、文書の内、部品化された文書に何の修正を加えることなくそのまま利用できる例えばビジネス文書、手引き書、手紙などのように構造化可能な文書を対象とし、これらの文書を部品化してデータベースに格納し、ユーザーが与える仕様に応じて必要な文書を作成することができるので、文書の作成の効率化を図ることができる。

【0053】特に、請求項2によれば、文書を部品化するに際し、文書のマークアップをデータベースの格納されている既存の部品化された文書の構造と比較しチェックできるので、マークアップの誤りを容易に発見することができる。請求項3によれば、部品を指定することにより、データベースよりその部品を用いた文書を自動的に検索し、一文書に合成することができるので、文書の作成を効率的に行うことができる。請求項4によれば、文書の出力フォーマットを入力することにより、データベースより必要な部品を検索し、指定した出力フォーマットの文書を自動的に生成できるので、文書の作成を効率的に行うことができるとともに、データベースに格納された1つの情報を出力フォーマットの異なった種々の文書で利用することができる。請求項5によれば、文書のタイプを入力することにより、入力された文書タイプの構造をデータベースより検索し、表示できるので、新しい文書を作成および文書の部品化を効率的に行うことができる。

#### 【図面の簡単な説明】

【図1】本発明の原理ブロック図である。

【図2】本発明における文書作成支援システムの全体構成図である。

【図3】マークアップ済文書の1例を示す図である。

【図4】文書構造の1例を示す図である。

【図5】データベース内の文書の格納形態を示す図である。

【図6】本発明の部品定義手段における構文チェック機能の実施例である。

【図7】本発明の部品定義手段における構文チェック機

能の説明図である。

【図8】本発明の部品定義手段における構文チェック機能のフローチャートである。

【図9】本発明の文書作成における第1の機能の実施例である。

【図10】本発明の文書作成における第1の機能の説明図である。

【図11】本発明の文書作成における第1の機能のフローチャートである。

【図12】本発明の文書作成における第2の機能の実施例である。

【図13】本発明の文書作成における第2の機能の説明図である。

【図14】ユーザが与える出力フォーマットの1例である。

【図15】文書合成されたマークアップ済文書の1例である。

【図16】出力文書の1例である。

【図17】出力フォーマット定義の構文木である。

【図18】本発明の文書作成における第2の機能のフローチャートである。

【図19】本発明における文書のデータ構造を表示する

機能の実施例である。

【図20】本発明における文書のデータ構造を表示する機能の説明図である。

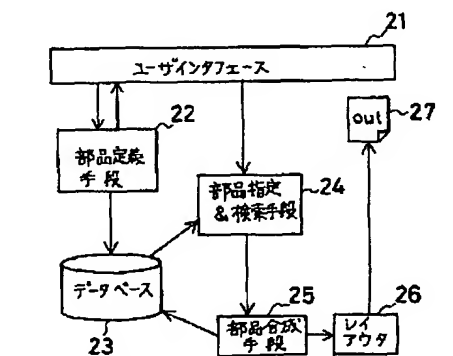
【図21】本発明における文書のデータ構造を表示する機能のフローチャートである。

【符号の説明】

1, 22	部品定義手段
2	部品指定&抽出手段
3, 25, 45, 56	部品合成手段
4, 23, 33, 47, 55, 64	データベース
5, 36	シンタックスチェック
6, 21, 31, 41, 51, 62	ユーザインタフェース
7, 43	部品解析手段
8, 44	部品抽出手段
9, 53	解析手段
10, 54	要素抽出手段
11, 63,	部品定義検索&抽出手段
12, 65	表示手段

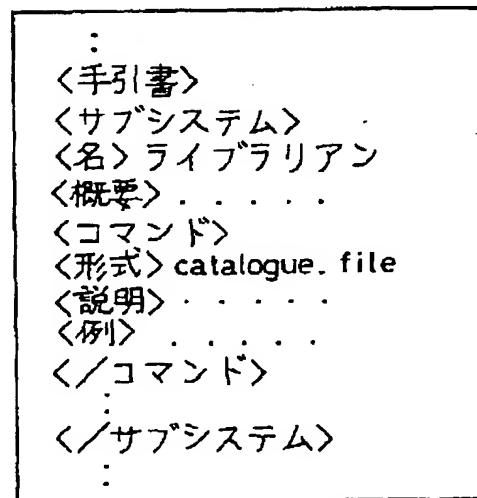
【図2】

本発明における文書作成支援システムの全体構成図



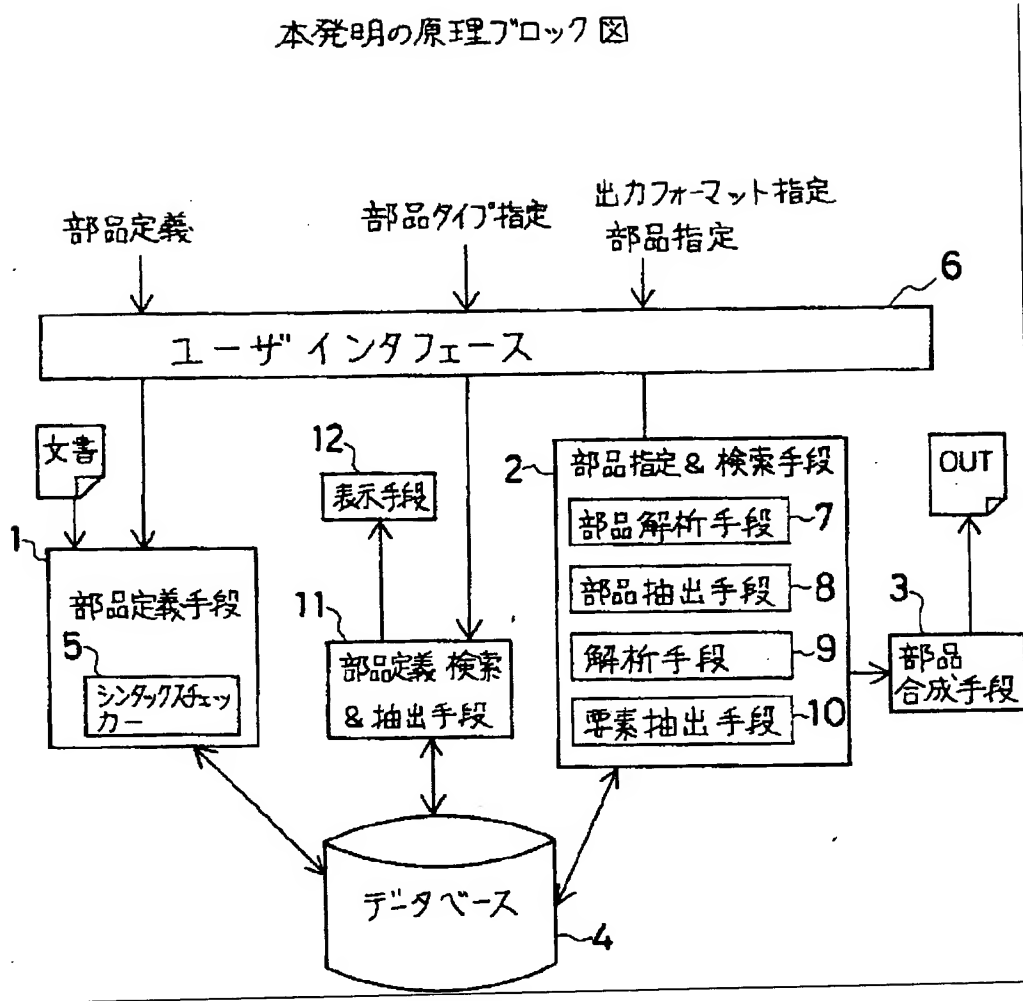
【図3】

マークアップ済文書の一例を示す図



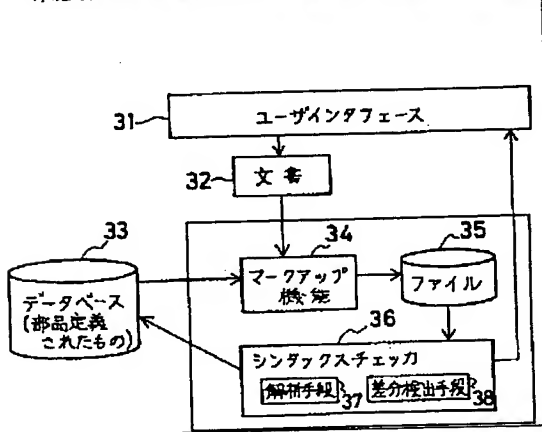
【図1】

本発明の原理ブロック図



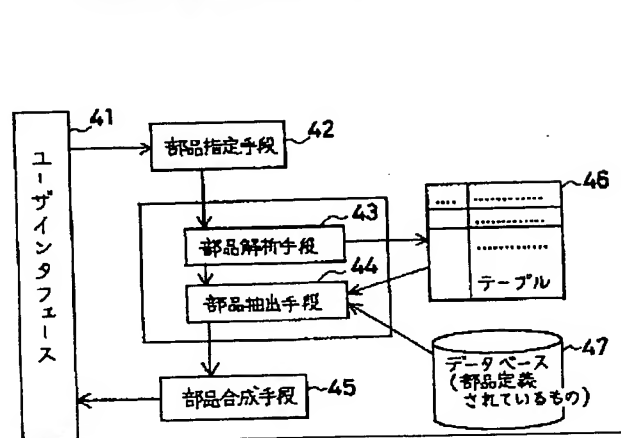
【図6】

本発明の部品定義手段における構文チェック機能の実施例



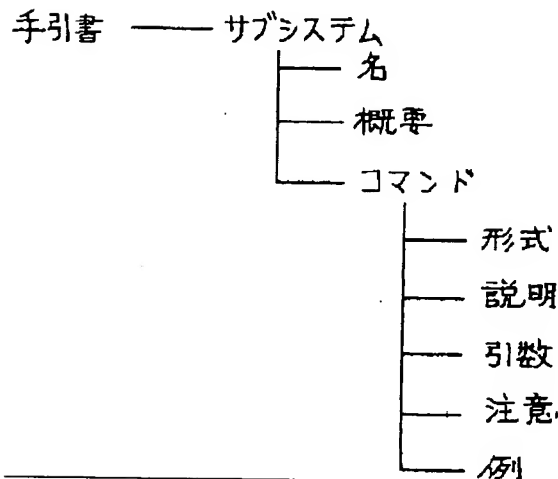
【図9】

本発明の文書作成における第1の機能の実施例



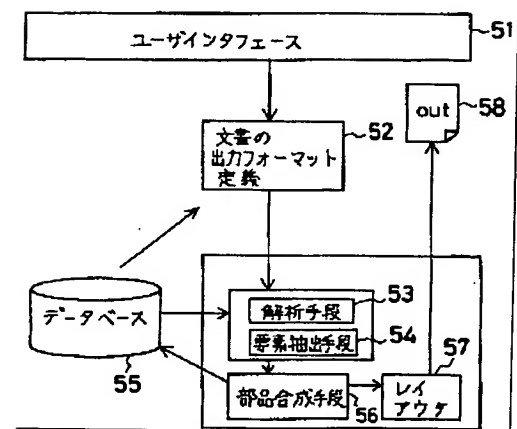
【図4】

文書構造の一例を示す図



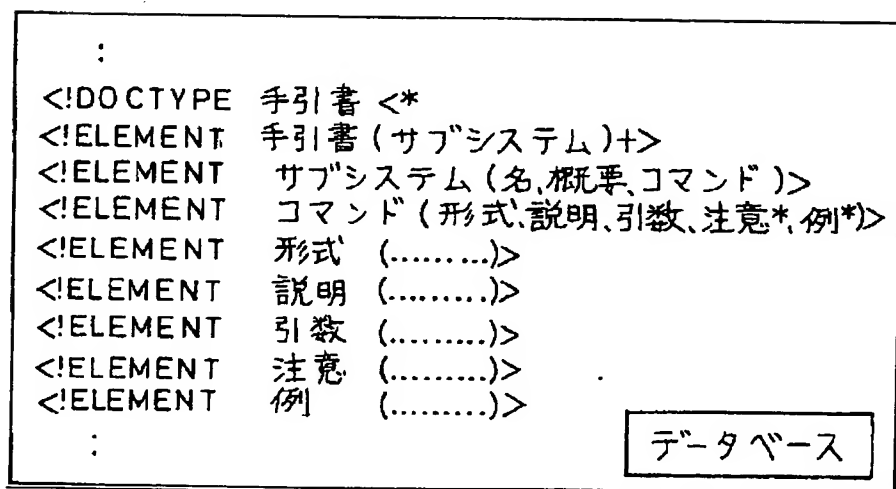
【図12】

本発明の文書作成における第2の機能の実施例



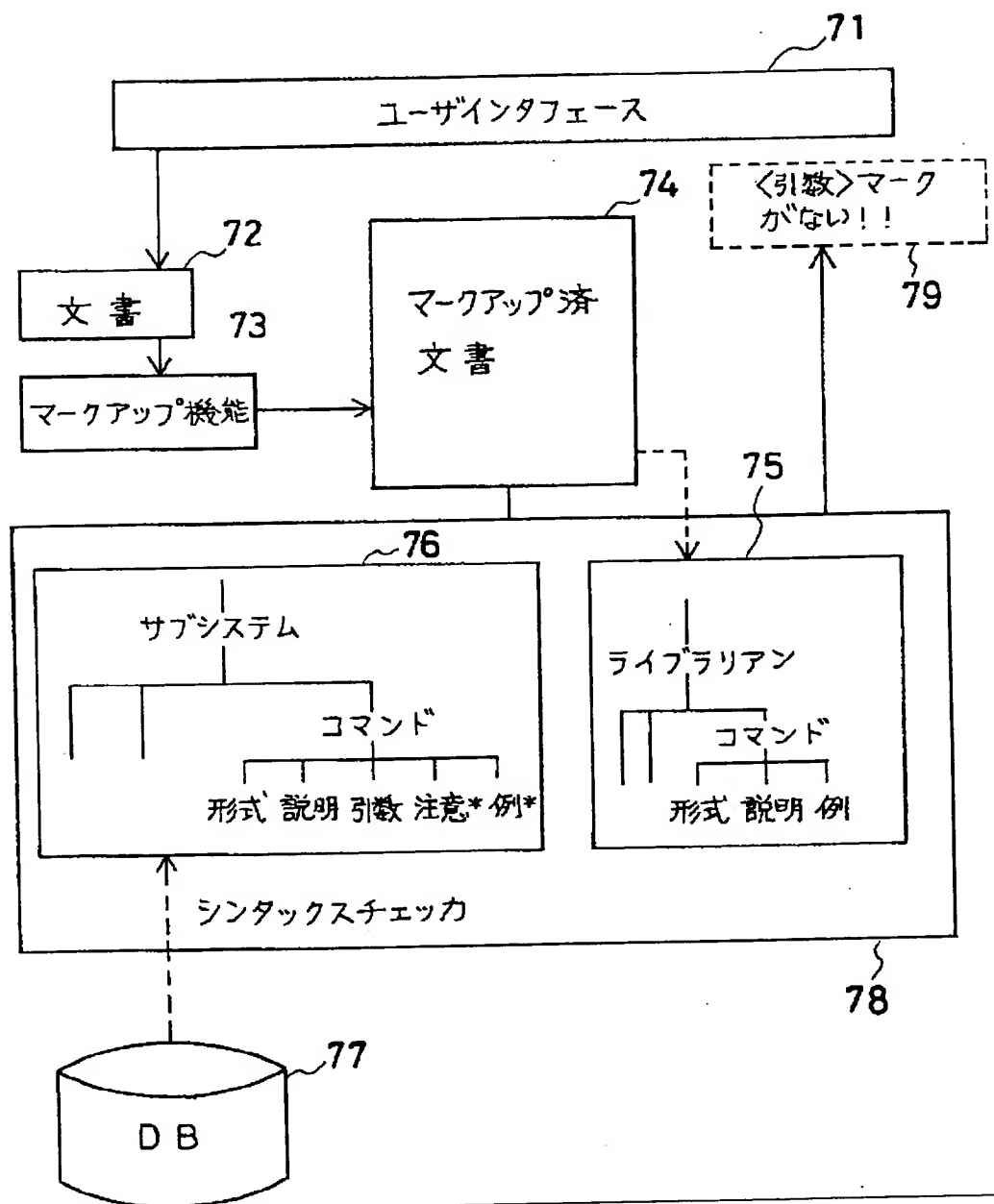
【図5】

データベース内の文書の格納形態を示す図



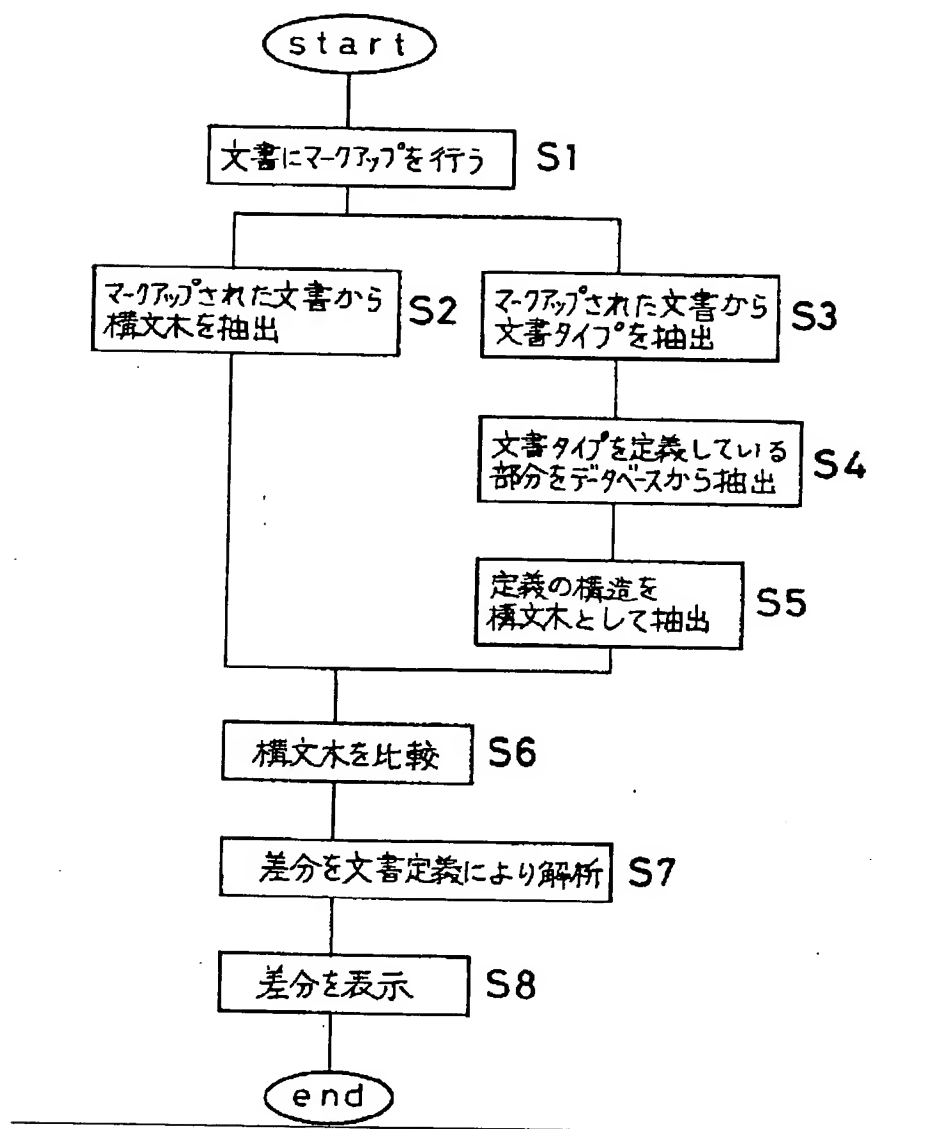
【図7】

本発明の部品定義手段における構文チェック機能の説明図



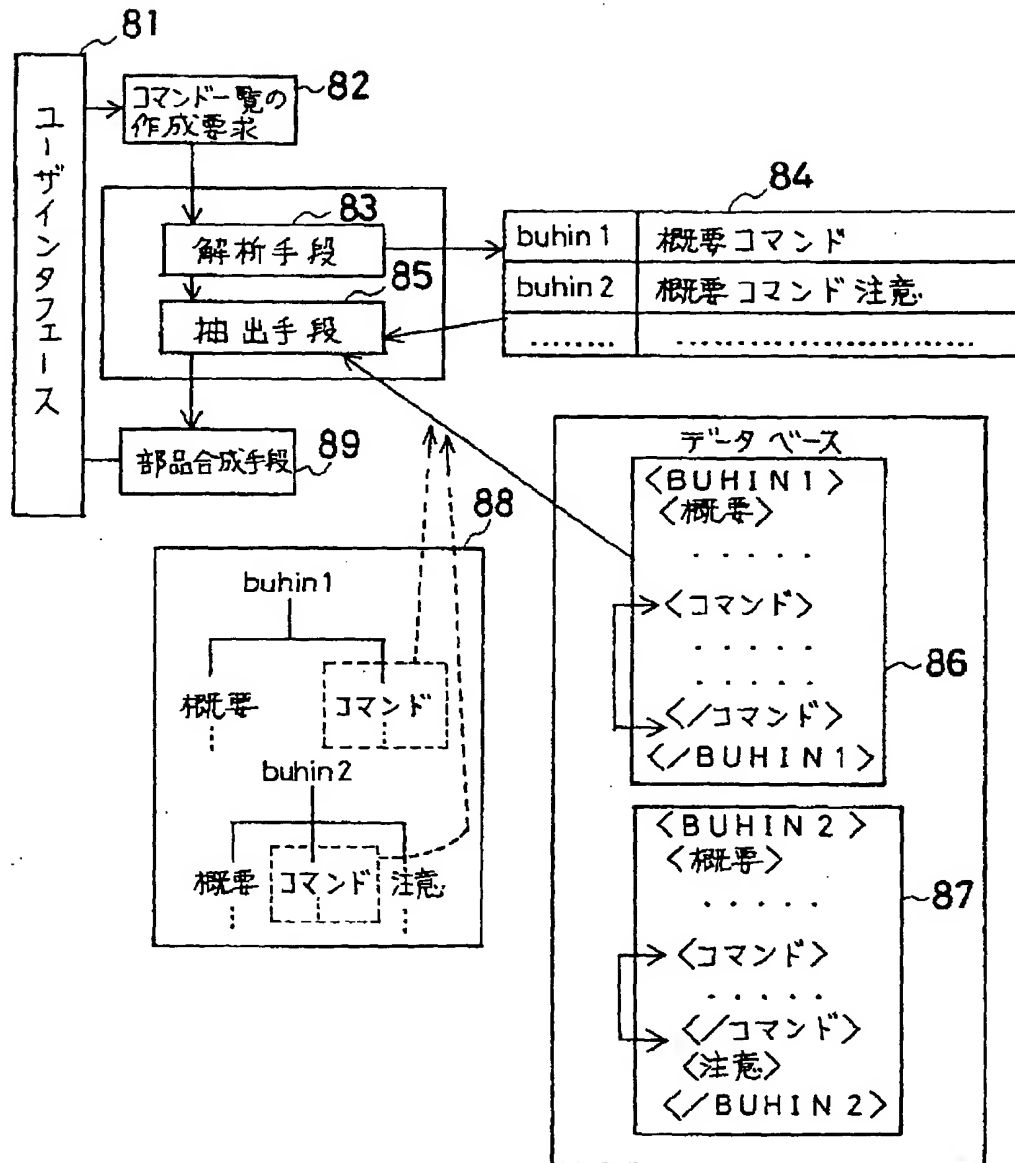
【図8】

本発明の部品定義手段における構文チェック機能のフローチャート



【図10】

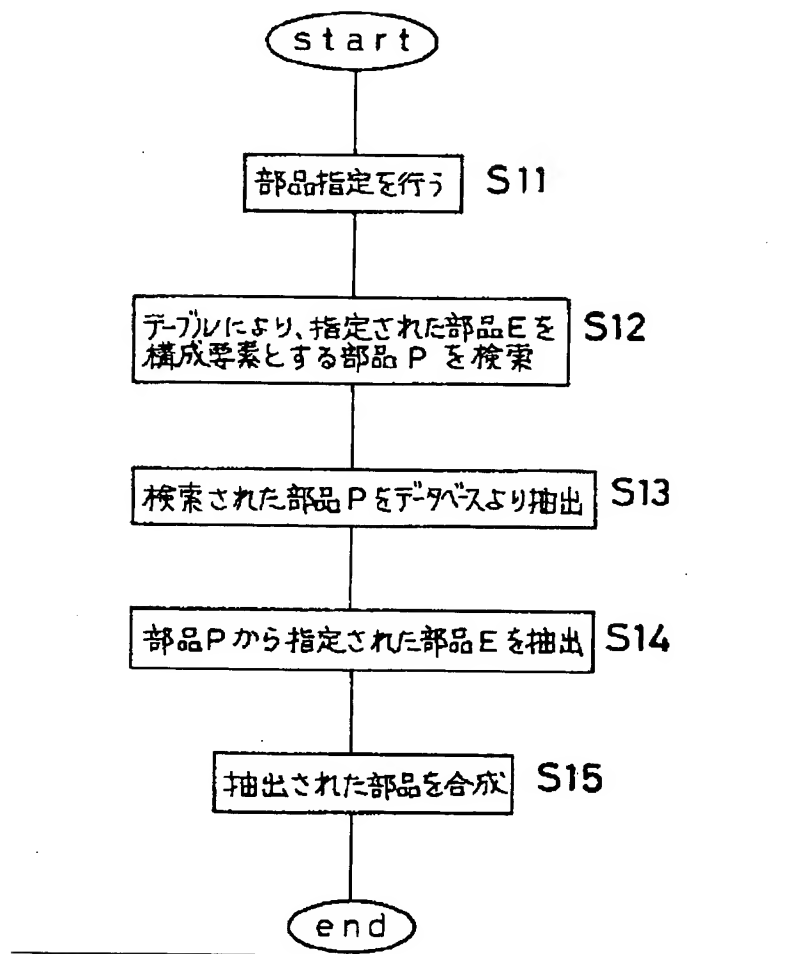
本発明の文書作成における第1の機能の説明図





【図11】

本発明の文書作成における第1の機能のフローチャート



【図14】

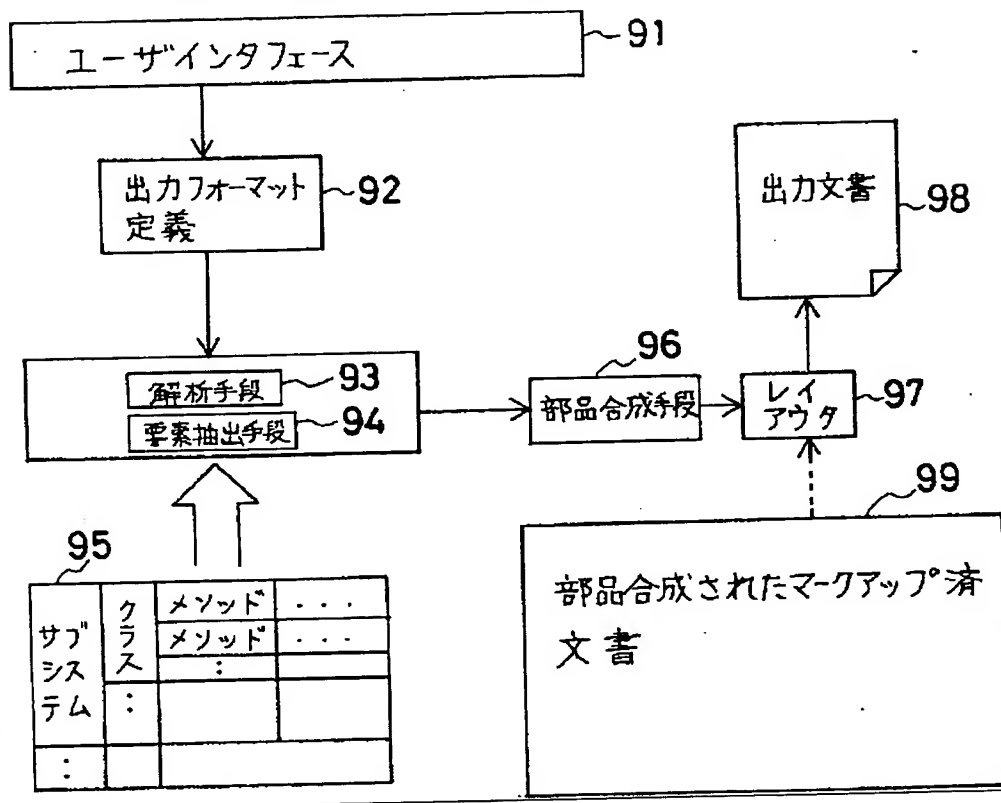
ユーザが与える出力フォーマットの1例

```

:
<!DOCTYPE ライブラリ <*>
<!ELEMENT ライブラリ (概要, サブシステム+)>
<!ELEMENT サブシステム (概要, クラス+)>
<!ELEMENT クラス (概要, メソッド+)>
<!ELEMENT メソッド (形式, 説明, 引数, 注意*, 例*)>
>>
  
```

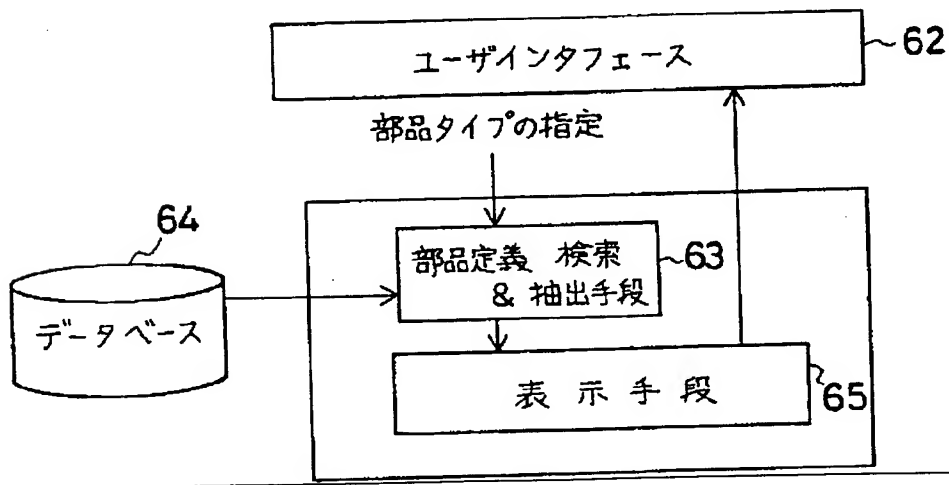
【図13】

本発明の文書作成における第2の機能の説明図



【図19】

本発明における文書のデータ構造を表示する機能の実施例



【図15】

文書合成されたマークアップ済文書の1例

```

<ライブラリ>
<概要> ライブラリの概要が書かれます。 ← a
<サブシステム> ライブラリアン ← b
<概要> ライブラリアンの概要が書かれます。 ← c
<クラス> class1
<概要> クラスの概要が書かれます。
<メソッド> method1
<形式> method1(a,b)
<説明> method1の説明です。
<引数> 引数a, 引数bの説明です。
<例> method1(Class,1)。使用例です。
</メソッド>
<メソッド>
    :: 次のメソッドの説明です。
</メソッド>
</クラス>
<クラス>
    :: 次のクラスの説明です。
</クラス>
</サブシステム>
<サブシステム>
    :: 別のシステムの説明です。
</サブシステム>
</ライブラリ>

```

【図16】

出力文書の1例

[概要]  
ライブラリの概要が書かれます。  
ライブラリアン

[概要]  
ライブラリアンの概要が書かれます。  
CLASS1

[概要] クラスの概要が書かれます。  
method1(A,B)

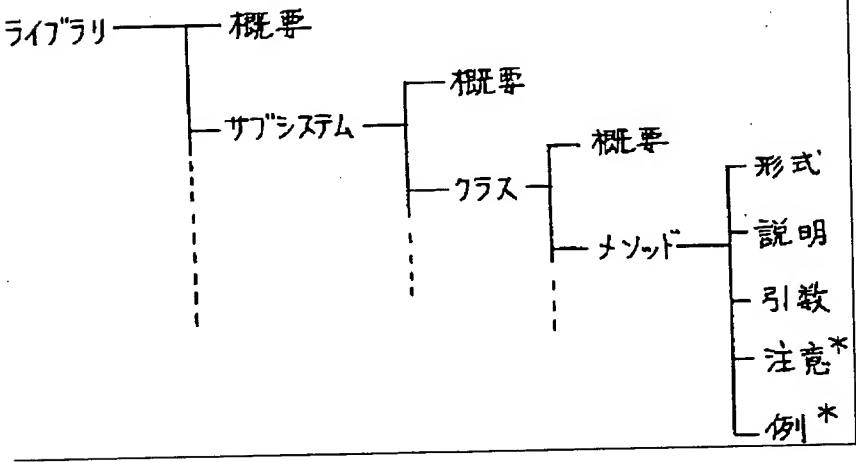
[形式] method1(a,b)

[説明] method1 の説明です。

[引数] 引数 a の説明です。  
引数 b の説明です。

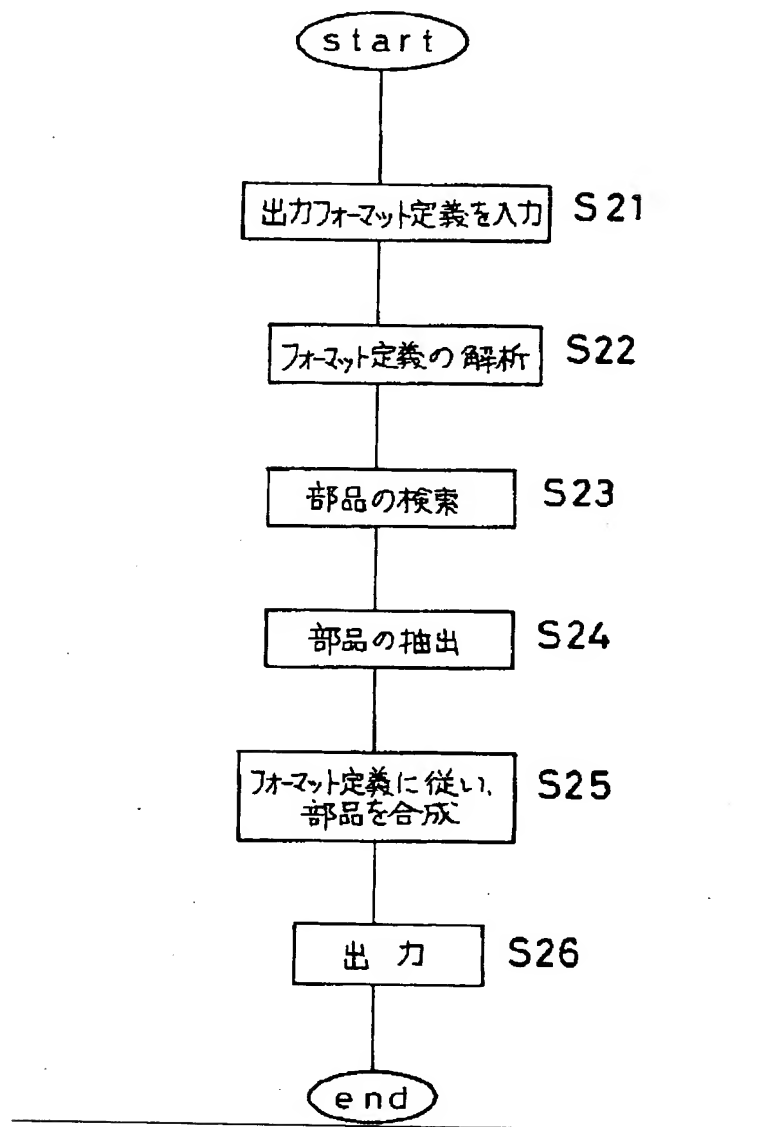
[例] method1(Class,1)。使用例です。  
::

出力フォーマット定義の構文木



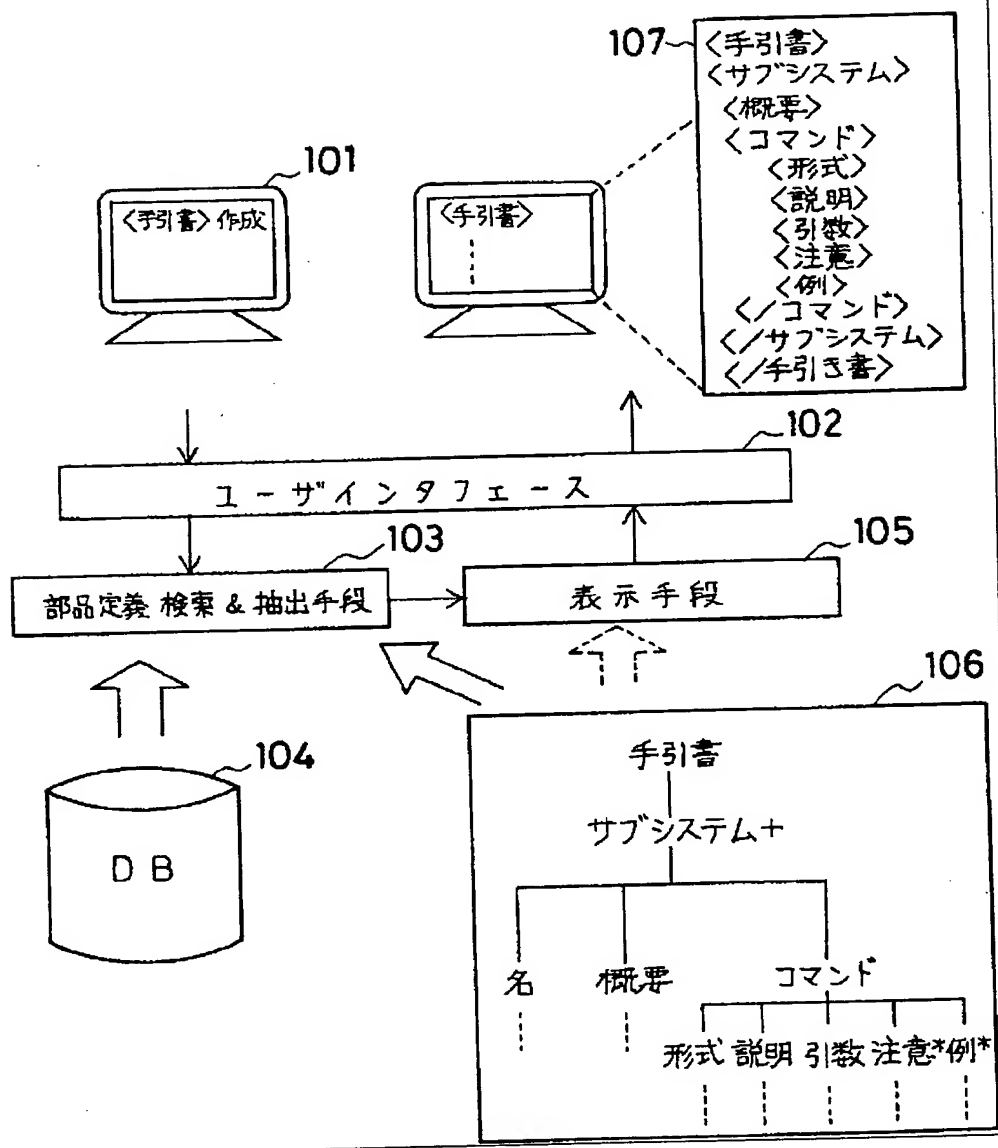
【図18】

本発明の文書作成における第 2 の機能のフローチャート



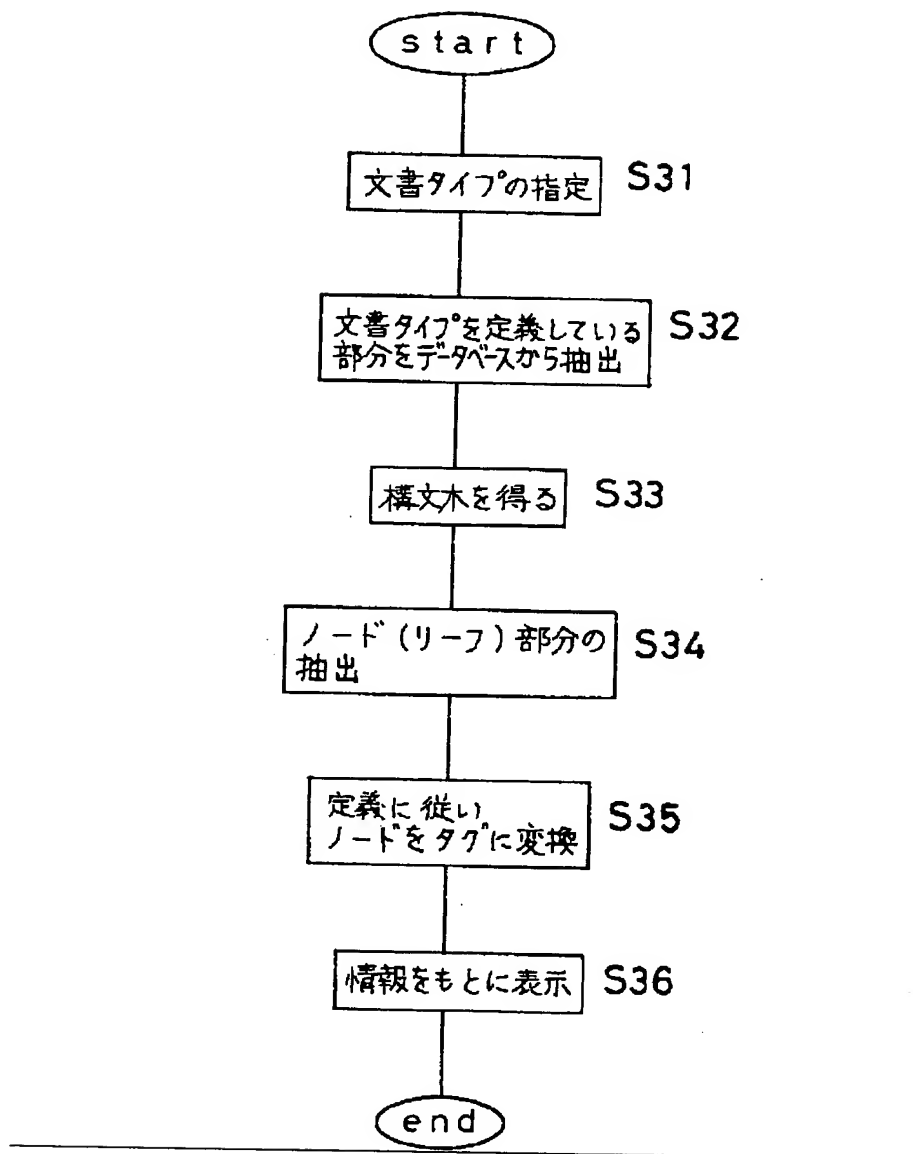
【図 20】

本発明における文書のデータ構造を表示する機能の説明図



【図21】

本発明における文書のデータ構造を表示する機能のフローチャート



【手続補正書】

【提出日】平成3年11月19日

【手続補正1】

【補正対象書類名】図面

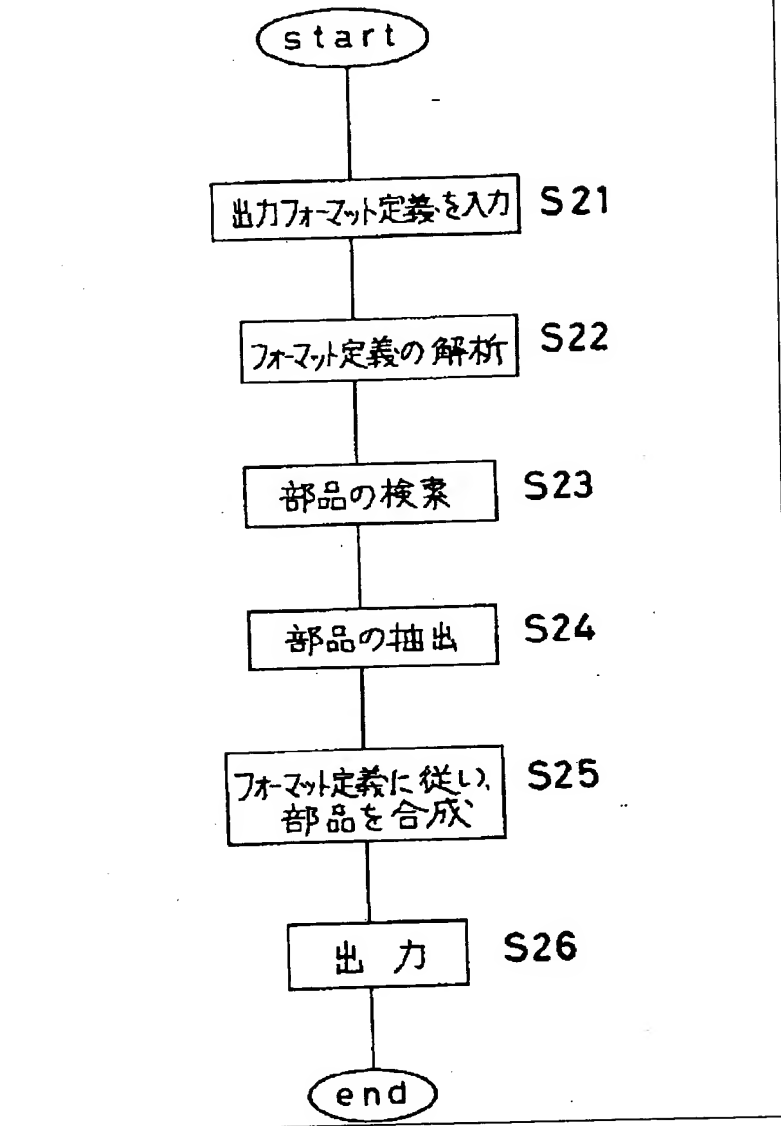
【補正対象項目名】図18

【補正方法】変更

【補正内容】

【図18】

## 本発明の文書作成における第2の機能のフローチャート



【手続補正書】

【提出日】平成3年12月25日

【手続補正1】

【補正対象書類名】図面

【補正対象項目名】図16

【補正方法】変更

【補正内容】

【図16】



## 出力文書の1例

[概要]	
ライブラリの概要が書かれます。	
<u>ライブラリアン</u>	
[概要]	
ライブラリアンの概要が書かれます。	
CLASS1	
[概要] クラスの概要が書かれます。	
method1(A,B)	
[形式]	method1(a,b)
[説明]	method1 の説明です。
[引数]	引数 a の説明です。 引数 b の説明です。
[例]	method1(Class,1)。使用例です。
::	

【手続補正2】

【補正対象書類名】図面

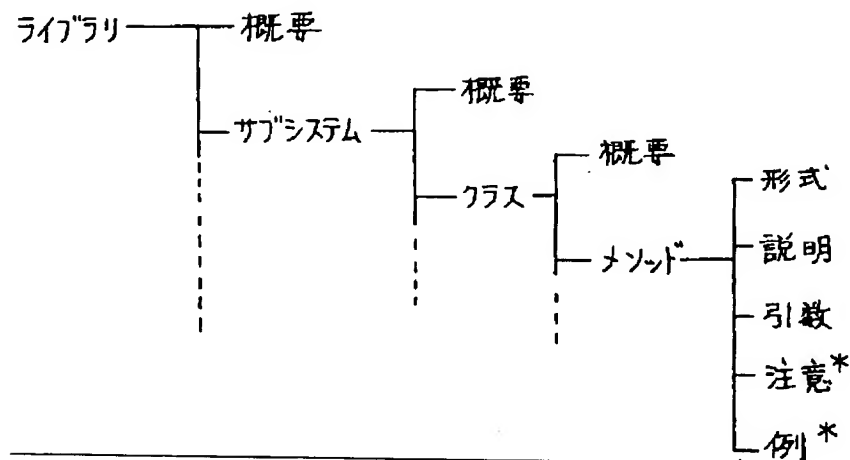
【補正対象項目名】図17

【補正方法】追加

【補正内容】

【図17】

## 出力フォーマット定義の構文木



10